Robust Optimization On Partial Network Design

Dimitri Hubans
dimitri.hubans@enac.fr
ENAC, Université de Toulouse
Toulouse, France

Sonia Cafieri
sonia.cafieri@recherche.enac.fr
ENAC, Université de Toulouse
Toulouse, France

Laurent Houssin
laurent.houssin@enac.fr
ENAC, Université de Toulouse
Toulouse, France

Abstract

This paper considers a variant of the network design problem where the available budget is not sufficient to satisfy all demands, and a choice must then be made regarding which demands are satisfied. We consider a robust version of this problem where the costs and the values of the demands are uncertain, and we introduce two variants of the min-max regret paradigm to address the problem. We propose both a decomposition algorithm and a column and constraint generation method and show through numerical experiments that the latter is the most efficient on the considered problem.

Keywords

Network design, Robust optimization, Min-max regret, Integer Linear Program, Column and constraint generation

1 Introduction

In network design problems, the objective is to build a network that satisfies a set of demands while minimizing a construction cost [11]. In practice, all the demands may not be satisfied because of some limited budget, and a choice is then to be made on what demand is to satisfy. A partial network design problem is then introduced. A known version of this problem is such that the number of satisfied demands is maximized while respecting a certain budget, a second one is such that the cost of the network is minimized while satisfying a minimum of demands. Both of these problems have been addressed and treated with a Benders decomposition in [4].

Another aspect that has to be taken into account in practical cases is the uncertainty on the parameters. Indeed, in practice, it can be hard to estimate precisely the costs or the exact demands. Typically, in the case of a city public transportation design, the demand between each pair origin/destination (O/D) may not be precisely known.

Robust optimization offers a variety of tools to mitigate the uncertainty by providing solutions that address all scenarios. Indeed, robust optimization has already been largely used by the network design optimization community. For example, it considers uncertain transportation costs [8], unreliable transportation lines or nodes [5], or uncertain sets of demand [6]. Besides, uncertainty has been considered under different forms such has polyhedral uncertainty [2], mixed integer set [6] and budget uncertainty set [5].

However, literature seems to be lacking studies regarding robust optimization approaches for the partial network covering problems. The goal of this paper is to address the partial network design with uncertainties in costs and demands. The problem therefore considered has uncertainty in both constraints and objective function.

INOC '26, Liège (Belgium)

When uncertainty concerns constraints, one usually seeks solutions that satisfy the constraints on all scenarios. Besides, multiple approaches exist within the framework of robust optimization to tackle uncertainty in the objective function. The contribution of this paper relies on the min-max regret paradigm. The usual definition of regret, as presented in [10], can be summarized as the difference between the resulting value of the decision versus the value of the solution that would have been taken if uncertain parameters would have been known in advance. Regret optimization has been regularly applied to problems in network design, see for example [6], [8]. Usually, the regret paradigm is applied only in problems where uncertainty is only on the objective function (see [10] for multiple examples). When constraints suffer from uncertainty, the definition of regret is slightly ambiguous. Indeed, two set of solutions can be considered for computing regret. It can either be the set of solutions robust to constraints or the set of solutions feasible in at least one scenario. Although both of these variants of regret have been used in the literature (see [6] and [1]), no study exists on the differences between these two regrets in the context of network design, to the best of our knowledge. We address the two notions of regret in the context of partial network design with uncertainty on the objective function and the constraints.

The remainder of the paper is organized as follows. First we present the underlying deterministic problem of partial network design in Section 2, then robust counterparts of this problem are described along with the different methods of resolution in Section 3. The proposed approaches are validated through experimental results in Section 4.

2 Problem definition

2.1 Deterministic version

The problem under consideration consists of a non-oriented graph $G = (V, E)$ where V is the set of vertices and E is the set of edges. Each edge $e = \{i, j\} \in E$ ($i, j \in V$) is associated with two different costs, a construction cost $c_e \geq 0$ (also referred to as build cost) and an use cost $u_e \geq 0$. The construction cost is paid once if the edge is used, and the use cost is paid for each use of the edge. For each edge $e = \{i, j\} \in E$ we define the arcs $a = (i, j), \hat{a} = (j, i)$, and we define A as the set of arcs of G built from E. For each vertex $i \in V$, we define $\delta^+(i) = \{(i, j) | (i, j) \in A\}$ the set of outgoing arcs of vertex i and $\delta^-(i) = \{(j, i) | (j, i) \in A\}$ the set of incoming arcs to i.

We consider a set W of demands on this graph. Each demand $w \in W$ is associated with an origin node w^s and a destination node w^t, with $w^s, w^t \in V$. Moreover, each demand has a certain gain $g_w \geq 0$. Finally a limited construction budget $C \geq 0$ is available, the total cost of the network (build cost plus use cost) cannot exceed this budget. The objective is then to maximize the gain of the designed network. This version of the partial network design problem is named the Maximal Network Covering Problem (MC).

To model this problem we consider the following set of variables:

10.48786/inoc.2026.12

- $x_e \in \{0,1\}, \forall e \in E$ with $x_e = 1$ if edge e is built in the solution (0 otherwise),
- $f_{wa} \in \{0,1\}, \forall a \in A, w \in W$ with $f_{wa} = 1$ if demand w goes through arc a (0 otherwise),
- $z_w \in \{0,1\}$ with $z_w = 1$ if demand w is satisfied (0 otherwise).

In the following, x denotes the vector of variables $(x_e)_{e\in E}$. Analogously, f denotes the vector of variable $(f_{wa})_{a\in A, w\in W}$ and z denotes the vector of variable $(z_w)_{w\in W}$.

An Integer Linear (ILP) formulation for the MC problem can be deduced from the variables:

(MC)

$$\max_{x,f,z} \sum_{w\in W} g_w z_w, \tag{1}$$

$$\sum_{e\in E} c_e x_e + \sum_{\substack{e=\{i,j\}\in E\\ a=(i,j),\hat{a}=(j,i)}} u_e \sum_{w\in W} (f_{wa} + f_{w\hat{a}}) \leq C, \tag{2}$$

$$\sum_{a\in\delta_w^+(i)} f_{wa} - \sum_{a\in\delta_w^-(i)} f_{wa} = \begin{cases} z_w, & \text{if } i = w^t, \\ -z_w, & \text{if } i = w^s, \\ 0 & \text{else,} \end{cases} \quad w \in W, i \in V, \tag{3}$$

$$f_{wa} \leq x_e, \qquad w \in W, a = (i,j) \in A : e = \{i,j\}, \tag{4}$$

$$x_e, z_w \in \{0,1\}, \qquad e \in E, w \in W, \tag{5}$$

$$f_a^w \in \{0,1\}, \qquad a \in A, w \in W. \tag{6}$$

The objective, expressed by (1) is to maximize the gain of the designed network. Constraint (2) represents the fact that the total cost of the network must not exceed the available budget of construction. Constraints (3) are flow constraints, ensuring that the flow is continuous between origin and destination of each demand and the flow is active only if the demand is covered. Constraints (4) ensure that the flow of each demand can only go through edges that are built in the design. Finally, constraints (5) and (6) are the binary constraints on the variables.

To make further models less cluttered, we denote by $\mathcal{F}$ the set of vectors (x, f, z) satisfying constraints (3) - (6).

It can be noted that constraints (6) can be relaxed to non-negativity constraints without changing the optimal value. Indeed, for any fixed value of x and z, the matrix representing the flow constraints is totally unimodular. Therefore a linear relaxation on the flow variable won't change the value of an optimal solution. The property of consecutive ones can be used to prove the total unimodularity (see for example [9]). However, in the uncertain variant of this problem the relaxation does not give the same value as the non-relaxed version.

A much related problem is the Partial Covering Network Design Problem (PC), in which a certain portion of the network has to be covered while minimizing costs. This problem is not treated in this paper, but the notions and solving methods introduced for problem MC can all be adapted for the PC problem.

2.2 Uncertainties

Optimization problems may suffer from uncertainty in the parameters due to measurement errors or lack of data. The idea behind robust optimization is to consider this uncertainty in the optimization problem. The uncertain parameters considered in the addressed network design problem are the gain associated to the demands and the costs of the edges.

Uncertainty sets are used to model the uncertain parameters. An uncertainty set represent the set of values of the uncertain parameters that are taken into account in the problem resolution. In the problem under consideration, two uncertainty sets are considered: $\mathcal{U}_g$ (resp. $\mathcal{U}_{cu}$) is the uncertainty set related to the gains (resp. the costs) parameters. A scenario is then a particular realization of one of these uncertainty set.

Uncertainty sets $\mathcal{U}_g$ and $\mathcal{U}_{cu}$ are defined by polyhedrons, that is by a set of linear constraints that the uncertain vector parameters must satisfy. g (resp. c, u) denotes the vector of parameters $(g_w)_{w\in W}$ (resp. $(c_e)_{e\in E}$, $(u_e)_{e\in E}$). The linear constraints associated to $\mathcal{U}_g$ and $\mathcal{U}_{cu}$ are noted respectively L and K. The uncertainty sets are then:

$$\mathcal{U}_g := \{g \geq 0 | \sum_{w\in W} a^g_{\ell w} g_w \leq b_\ell, \forall \ell \in L\} \tag{7}$$

and

$$\mathcal{U}_{cu} := \{(c,u) \geq 0 | \sum_{e\in E} a^c_{ke} c_e + a^u_{ke} u_e \leq b_k, \forall k \in K\}. \tag{8}$$

These uncertainty sets are very general, as they can model a variety of technical constraints and are a generalization of the often used budget of uncertainties introduced in [3].

A common approach in robust optimization is to find a solution that is feasible in all scenarios. In the considered MC problem, the feasibility of a solution will depend on the scenario $(c,u) \in \mathcal{U}_{cu}$. Therefore, we denote by $\mathcal{F}_{c,u}$ the set of solutions $(x,f,z) \in \mathcal{F}$ satisfying:

$$\sum_{e\in E} c_e x_e + \sum_{\substack{e=\{i,j\}\in E\\ a=(i,j),\hat{a}=(j,i)}} u_e \sum_{w\in W} (f_{wa} + f_{w\hat{a}}) \leq C.$$

The set of solutions that are feasible in all scenarios is therefore $\mathcal{F}^\cap = \cap_{(c,u)\in\mathcal{U}_{cu}} \mathcal{F}_{c,u}$. This set is called the set of *robust* solutions. Therefore, any robust solution must satisfy the following constraint:

$$\max_{(c,u)\in\mathcal{U}_{cu}} \left(\sum_{e\in E} c_e x_e + \sum_{\substack{e=\{i,j\}\in E\\ a=(i,j),\hat{a}=(j,i)}} u_e \sum_{w\in W} (f_{wa} + f_{w\hat{a}}) \right) \leq C. \tag{9}$$

The left hand side of constraint (9) is the objective function value of the following linear program:

$(WCSP(x,f))$

$$\max_{c,u} \sum_{e\in E} x_e c_e + \sum_{\substack{e=\{i,j\}\in E\\ a=(i,j),\hat{a}=(j,i)}} u_e \sum_{w\in W} (f_{wa} + f_{w\hat{a}}), \tag{10}$$

$$\sum_{e\in E} a^c_{ke} c_e + a^u_{ke} u_e \leq b_k, \qquad k \in K, \tag{11}$$

$$c_e, u_e \geq 0, \qquad e \in E \tag{12}$$

Where constraints (11) and (12) are equivalent to $(c,u) \in \mathcal{U}$.

By strong duality the left part of (9) is also the optimal value of the dual linear program:

$$(D\_WCSP(x,f))$$
$$\min_q \sum_{k\in K} b_k q_k, \tag{13}$$
$$\sum_{k\in K} a^c_{ke} q_k \geq x_e, \qquad e\in E, \tag{14}$$
$$\sum_{k\in K} a^r_{ke} q_k \geq \sum_{w\in W} f_{wa} + f_{w\hat{a}}, \qquad e\in E. \tag{15}$$
$$q_k \geq 0 \qquad k\in K \tag{16}$$

Therefore a solution design $(x, f, z) \in \mathcal{F}$ is robust if and only if there exists a positive vector q such that:

$$\sum_{k\in K} b_k q_k \leq C, \tag{17}$$
$$+(14), (15), (16). \tag{18}$$

Thus, in the remaining models of the paper, having $(x, f, z) \in \mathcal{F}^{\cap}$ implies only a number of linear constraints polynomial within the size of the problem.

3 Robust regret approaches

We focus on the min-max regret paradigm. As mentioned in section 1, two ways to interpret the notion of regret can be considered for the network design problem under consideration. The first one is to compute the regret to solutions that are also robust to constraints and the second one is to compute the regret to solutions feasible only in a fully deterministic setting. An interpretation of these two notions can be given as follows. The later represents the regret of not having known all the uncertain parameters prior to the optimization process. By contrast, the other one is a regret to solutions that were reasonable to take for the decision maker in the uncertain settings.

As far as we know, the differences between the two notions of regret have not been explored in the literature. We name the two regrets pessimistic and optimistic regrets (OR and PR). In the MC problem, the optimistic regret of a solution (x, f, z) is

$$R_O(x,f,z) := \max_{g\in\mathcal{U}_g}\left(Z_O(g) - \sum_{w\in W} g_w z_w\right). \tag{19}$$

Where $Z_O(g)$ is the optimal value of:

$$\max_{x,f,z} \sum_{w\in W} g_w z_w, \tag{20}$$
$$\min_{(c,u)\in\mathcal{U}_{cu}}\left(\sum_{e\in E} c_e x_e + \sum_{\substack{e=\{i,j\}\in E\\ a=(i,j),\hat{a}=(j,i)}} u_e \sum_{w\in W}(f_{wa}+f_{w\hat{a}})\right) \leq C, \tag{21}$$
$$(x,f,z)\in\mathcal{F}. \tag{22}$$

By contrast, the pessimistic regret is defined by:

$$R_P(x,f,z) := \max_{g\in\mathcal{U}_g}\left(Z_P(g) - \sum_{w\in W} g_w z_w\right). \tag{23}$$

Where $Z_P(g)$ is the optimal value to:

$$\max_{x,f,z} \sum_{w\in W} g_w z_w, \tag{24}$$
$$\max_{(c,u)\in\mathcal{U}_{cu}}\left(\sum_{e\in E} c_e x_e + \sum_{\substack{e=\{i,j\}\in E\\ a=(i,j),\hat{a}=(j,i)}} u_e \sum_{w\in W}(f_{wa}+f_{w\hat{a}})\right) \leq C, \tag{25}$$
$$(x,f,z)\in\mathcal{F}. \tag{26}$$

A scenario and a solution maximizing the regret (pessimistic or optimistic) of a given solution are named *adversarial* scenario and solution.

Since the objective function of the deterministic problem only depends on z, the regrets (pessimistic or optimistic) of a solution also depends only on z. An adversarial scenario and solution and the optimistic regret of any solution $\hat{z}$ can be computed by solving the following MILP:

$$(ORSP(\hat{z}))$$
$$\max_{x,f,z,c,u,g} \sum_{w\in W} g_w(z_w - \hat{z}_w), \tag{27}$$
$$\sum_{e\in E} c_e x_e + \sum_{\substack{e=\{i,j\}\in E\\ a=(i,j),\hat{a}=(j,i)}} u_e \sum_{w\in W}(f_{wa}+f_{w\hat{a}}) \leq C, \tag{28}$$
$$\sum_{w\in W} a^g_{\ell w} g_w \leq b_\ell, \tag{29}$$
$$+\ (11), \tag{30}$$
$$(x,f,z)\in\mathcal{F},$$
$$c_e, u_e, g_w \geq 0, \qquad w\in W, e\in E. \tag{31}$$

This model is not linear due to the multiple products of variables ($g_w z_w$, $c_e x_e$ and $u_e(\sum_{w\in W} f_{wa} + f_{w\hat{a}})$). Products of binary and continuous variables can however easily be linearized using standard techniques.

Similarly, the following MILP gives the pessimistic regret of a solution $\hat{z}$ along with an adversarial scenario and solution:

$$(PRSP(\hat{z}))$$
$$\max_{x,f,z,g} \sum_{w\in W} g_w(z_w - \hat{z}_w), \tag{32}$$
$$+\ (29),$$
$$(x,f,z)\in\mathcal{F}^{\cap}, \tag{33}$$
$$g_w \geq 0, \qquad w\in W. \tag{34}$$

Once again, the products of variables $g_w z_w$ can be linearized using standard techniques.

The two previous models give means to compute the optimistic and pessimistic regret of any solution. However the goal is to find solutions minimizing their maximum regrets. This can be achieved by the following models:

$$\min_{x,f,z}\max_{g\in\mathcal{U}_g}\left(Z_P(g) - \sum_{w\in W} g_w z_w\right), \tag{35}$$
$$(x,f,z)\in\mathcal{F}^{\cap}. \tag{36}$$

and

$$\min_{x,f,z} \max_{g\in\mathcal{U}_g} \left(Z_O(g) - \sum_{w\in W} g_w z_w \right), \quad (37)$$

$$(x,f,z) \in \mathcal{F}^{\cap}. \quad (38)$$

It can be noticed here that the set of solution that are considered acceptable is the same for both problem. Indeed, regardless of the paradigm chosen for the objective function we always want to obtain a solution robust to the uncertain constraints, therefore belonging in $\mathcal{F}^{\cap}$.

The two models can respectively be linearized as:

$(PRMC)$

$$\min_{x,f,z,R} R \quad (39)$$

$$R \geq Z_P(g) - \sum_{w\in W} g_w z_w, \qquad g \in \mathcal{U}_g, \quad (40)$$

$$(x,f,z) \in \mathcal{F}^{\cap}. \quad (41)$$

and

$(ORMC)$

$$\min_{x,f,z,R} R \quad (42)$$

$$R \geq Z_O(g) - \sum_{w\in W} g_w z_w, \qquad g \in \mathcal{U}_g, \quad (43)$$

$$(x,f,z) \in \mathcal{F}^{\cap}. \quad (44)$$

Constraints (43) and (40) written this way imply an infinite number of constraints. However, because $(ORSP)$ and $(PRSP)$ are mixed-integer linear programs, it suffices to consider only the extreme points (vertices) of the associated polyhedron to represent the entire feasible set. Note that, even by doing so, the implied number of constraints would be exponential in the size of the problem.

To address this issue, two different types of decomposition can be considered, both of which can be applied to both regrets.

The first one follows a classical decomposition scheme. According to this scheme, the master problems are $(ORMC)$ and respectively $(PRMC)$ but with constraints (43) and respectively (40) relaxed by considering subsets of scenarios instead of $\mathcal{U}_g$. The sub-problems are $(ORSP)$ and respectively $(PRSP)$.

After each solving of the master problem, the solution obtained is evaluated through the regret sub-problem ($ORSP$ or $PRSP$) and the adversarial scenario maximizing the regret of the current solution is added to the master problem along with the optimal value that can be reached in this scenario. A new cut is then generated. Moreover, upper bounds for the problem can be updated through the objective function value of the master problem and lower bounds can be updated through the value of the objective function of the sub-problem. Cuts are generated and bounds are updated until the gap between lower bound and upper bound is small enough. Algorithm 1 describes this decomposition and is the same for both regret problems: MP refers to $ORMC$ or $PRMC$ while $RSP(z)$ refers to $ORSP(z)$ or $PRSP(z)$. LB and UB denote the lower and upper bound on the objective function value of MP.

Algorithm 1 Decomposition method

$LB := 0$; $UB := \infty$.
$(x^*, f^*, z^*) := 0$.
while $UB - LB \geq 0$ **do**
  Solve MP, let $(\hat{x}, \hat{f}, \hat{z}, R)$ be the obtained solution.
  $LB := R$.
  Solve $RSP(z)$ and retrieve g and $Z(g)$.
  Add $R \geq Z(g) - \sum_{w\in W} g_w z_w$ to MP.
  if $Z(g) < UB$ **then**
    $UB := Z(g)$.
    $(x^*, f^*, z^*) := (\hat{x}, \hat{f}, \hat{z})$.
  end if
end while
Return (x^*, f^*, z^*).

3.1 A column and constraint generation algorithm

The previous decomposition method can significantly be improved. Indeed the method can benefit from the dualization of the inner constraints:

$$R \geq Z(g) - \sum_{w\in W} g_w z_w, \qquad \forall g \in \mathcal{U}_g, \quad (45)$$

$$\Leftrightarrow R \geq \max_{g\in\mathcal{U}_g} \left(Z(g) - \sum_{w\in W} g_w z_w \right), \quad (46)$$

$$\Leftrightarrow R \geq \max_{g\in\mathcal{U}_g} \left(\max_{\hat{z}\in\mathcal{D}} \sum_{w\in W} g_w \hat{z}_w - \sum_{w\in W} g_w z_w \right), \quad (47)$$

$$\Leftrightarrow R \geq \max_{\hat{z}\in\mathcal{D}} \max_{g\in\mathcal{U}_g} \left(\sum_{w\in W} g_w \hat{z}_w - \sum_{w\in W} g_w z_w \right), \quad (48)$$

$$\Leftrightarrow R \geq \max_{g\in\mathcal{U}_g} \left(\sum_{w\in W} g_w \hat{z}_w - \sum_{w\in W} g_w z_w \right), \qquad \forall \hat{z} \in \mathcal{D}, \quad (49)$$

with $\mathcal{D}$ the domain of possible values of z depending on the type of regret considered. For pessimistic regret, $\mathcal{D} = \cap_{(c,u)\in\mathcal{U}_{cu}} \mathcal{F}_{c,u} = \mathcal{F}^{\cap}$. For optimistic regret, $\mathcal{D} = \cup_{(c,u)\in\mathcal{U}_{cu}} \mathcal{F}_{c,u} = \mathcal{F}^{\cup}$.

The expression $\max_{g\in\mathcal{U}_g} \left(\sum_{w\in W} g_w \hat{z}_w - \sum_{w\in W} g_w z_w \right)$ corresponds to the optimal value of a linear program, and can therefore be dualized.

The dual depends on $\hat{z}$ and can be written as follows:

$$\min_h \sum_{\ell\in L} b_\ell h_\ell, \quad (50)$$

$$\sum_{\ell\in L} a^g_{w\ell} h_\ell \leq \hat{z}_w - z_w, \qquad w \in W, \quad (51)$$

$$h_\ell \geq 0, \qquad \ell \in L. \quad (52)$$

h are the dual variables associated to constraints to (11) and constraints (51) are the dual constraints associated to variables g.

Constraint (49) can therefore be linearized as:

$$R \geq \sum_{\ell\in L} b_\ell h^{\hat{z}}_\ell, \qquad \hat{z} \in \mathcal{D}, \quad (53)$$

$$\sum_{\ell\in L} a^g_{w\ell} h^{\hat{z}}_\ell \leq \hat{z}_w - z_w, \qquad w \in W, \hat{z} \in \mathcal{D}, \quad (54)$$

$$h^{\hat{z}}_\ell \geq 0, \qquad \ell \in L, \hat{z} \in \mathcal{D}. \quad (55)$$

As a result of the above reformulation of constraint (43) and (40), integrating it in Algorithm 1 within the decomposition-based algorithm results in the addition of columns and constraints. At each step, we add the cut associated to an adversary solution instead of the cut associated to an adversary scenario.

As mentioned above, this columns and constraints generation method is suited for optimizing both regrets: $\mathcal{D}$ is replaced by $\mathcal{F}^{\cap}$ for pessimistic regret or $\mathcal{F}^{\cup}$ for optimistic regret.

4 Tests and results

4.1 Testing environment

Table 1 defines the notation used in the comparison of the two decomposition approaches on the two versions of the regret problem.

	OR	PR
Decomposition	Dec_ OR	Dec_PR
Col. and const. gen.	CCG_OR	CCG_PR

Table 1: Methods name

The considered methods have been tested on 60 instances with 6 different number of nodes, the arcs are generated randomly. The different number of nodes considered are $\{9, 12, 15, 16, 20, 25\}$. Instances are available at https://github.com/DHubans/Instances-robust-partial-network-design

All tests have been conducted on an i5 intel processor with 8 cores using the python interface of Gurobi solver version 12.0.3. All solver's parameters were set to default.

4.2 Results

To compare the different methods and paradigms, we use the performance profile method introduced in [7]. A performance profile plot is a cumulative distribution graph that shows, for each algorithm, the proportion of problem instances it solves within a given factor of the best observed performance.

Figure 1 shows the performance profiles of the 2 decomposition methods applied on the the two versions of the problem, with the solving time as performance evaluation. The plots are constructed as follows: for each instance i and each method applied on each problem $m \in \{Dec\_OR, Dec\_PR, \_CCG\_OR, CCG\_PR\}$ the time needed to solve instance i with method m is noted $t_{m,i}$. For each instance i the best solving time is $t_i^* := \min_m t_{m,i}$. Then for each instance i and method m a performance ratio is given by $\tau_{m,i} := \frac{t_{m,i}}{t_i^*}$. The x-axis of the plots are then performance ratios τ and the y-axis are the proportion of methods with a performance ratio better than τ. If an instance i is not solved with method m within the time limit given (1800 seconds), than its performance ratio is set at $\tau_{\max} = 75$. We can see here that the classical decompositions are much less efficient than the column and constraint generation algorithms. Therefore, the column and constraint generation-based decomposition algorithm is used in the following.

We now explore the behaviour of the pessimistic and optimistic regret paradigms.

Figure 1 points out that CCG_OR is the fastest method for 42 % of the instances while CCG_PR is the fastest method for only 30 % of the instances. However, by looking at the last values of τ below $\tau_{\max}$, 80 % of the instances are solved by CCG_PR within the time limit while 78 % of the instances were solved by

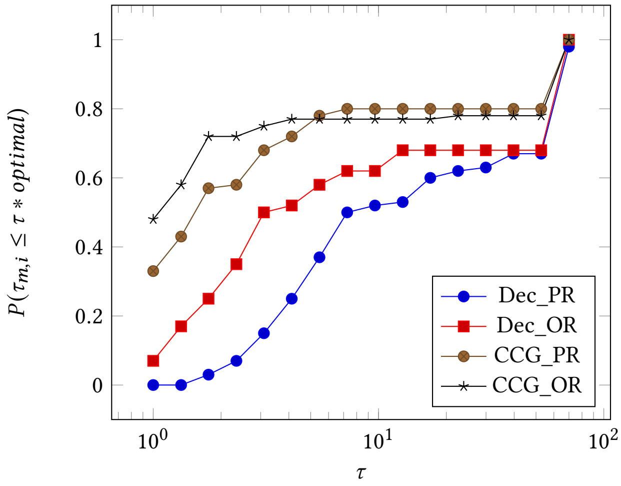

Figure 1: Solving time performance profile

CCG_OR within the time limit. This 2 % gap means that only one more instance is solved by CCG_PR compared to CCG_OR. Figure 2 gives a more precise description of the solving time of each instance by CCG_OR and CCG_PR. Each dot represents one instance and the dotted line represents the time limit. It can be noted that indeed 3 instances are solved to optimality by CCG_PR but not by CCG_OR and 2 instances are solved to optimality by CCG_OR but not by CCG_PR. Therefore, it is hard to conclude which regret is more difficult to optimize since most of the instances have a very close solving time whether for the OR of for the PR.

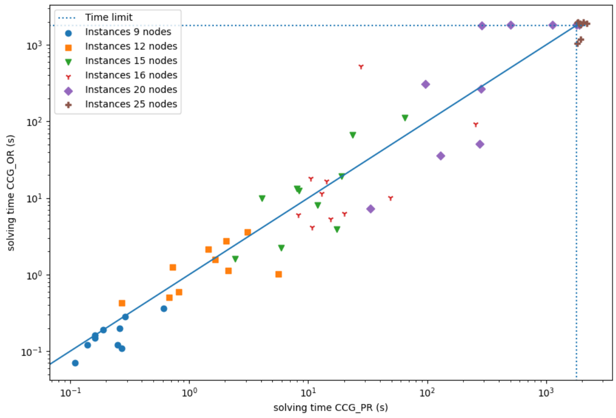

Figure 2: CCG_OR and CCG_PR solving times

To compare the pessimistic and optimistic paradigms, we now examine the solutions obtained at the end of the optimization process. Figure 3 represents the performance profile with optimistic regret as a performance metric. Only the instances optimally solved by both regrets are considered to compute the performance ratio. The figure is to be interpreted as follows: the constant value of 0.75 for CCG_OR means that 75% of the instances were optimally solved by both methods. Besides, value of CCG_OR at $\tau = 1$ indicates that for 28% of the instances the computed solutions have the same corresponding value of optimistic regret. It can also be noted that there is at most a factor of 1.15 between the optimistic regret value obtained with the OR problem and the one of the PR problem

Comparatively, on Figure 4 the relative gap between the solutions of CCG_PR and CCG_OR in terms of pessimistic regret value is larger. We remark that for 10% of the instances the solution of CCG_RO is worse than the one of CCG_RP by a factor at least 1.5. Moreover, in the worst case, the solution of CCG_RO is worse than the one of CCG_RP by a factor almost 2.

Therefore, it seems that solutions minimizing the pessimistic regret are also good regarding the optimistic regret, conversely it does not seem to be the case for solution optimizing the optimistic regret and their pessimistic regret value.

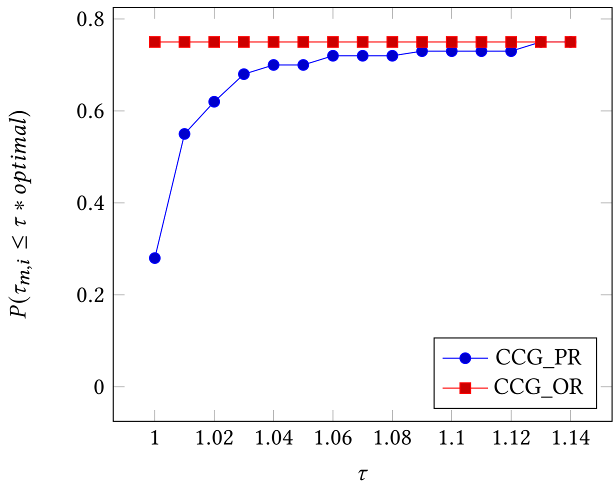

Figure 3: Optimistic regret performance profile

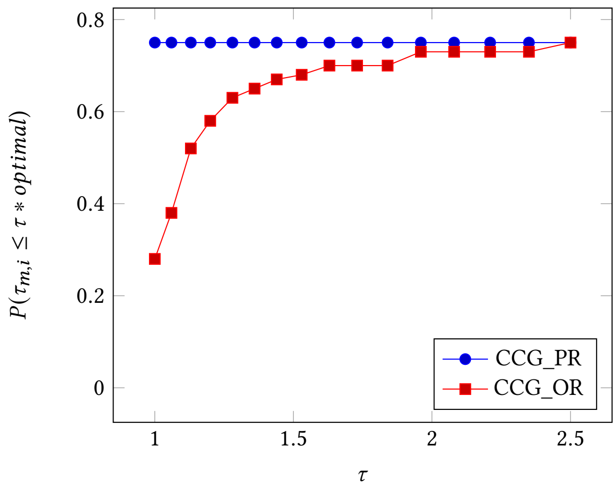

Figure 4: Pessimistic regret performance profile

5 Conclusion

In this paper, we addressed a robust network design problem, investigating two formulations of the problem relying on two different notions of regret. We presented two methods of resolution for the two problems and compared their performance on a set of instances. It appears that the constraints and column generation approach is much more efficient than the decomposition approach. Comparing the behaviour of the two regret paradigms, we remark that the solutions obtained with the two regrets are different, the solution obtained with pessimistic regret perform also well for the optimistic regret, but that the solution obtained with optimistic regret have a significantly higher pessimistic regret than the optimal one.

References

[1] Necati Aras and Ümit Bilge. 2018. Robust supply chain network design with multi-products for a company in the food sector. *Applied Mathematical Modelling* 60 (2018), 526–539.

[2] Walid Ben-Ameur and Hervé Kerivin. 2005. Routing of uncertain traffic demands. *Optimization and Engineering* 6, 3 (2005), 283–313.

[3] Dimitris Bertsimas and Melvyn Sim. 2004. The Price of Robustness. *Operations Research* 52 (02 2004).

[4] Víctor Bucarey, Bernard Fortz, Natividad González-Blanco, Martine Labbé, and Juan A. Mesa. 2022. Benders decomposition for network design covering problems. *Computers & Operations Research* 137 (2022).

[5] Chun Cheng, Mingyao Qi, Ying Zhang, and Louis-Martin Rousseau. 2018. A two-stage robust approach for the reliable logistics network design problem. *Transportation Research Part B: Methodological* 111 (2018).

[6] Eduardo Conde and Marina Leal. 2021. A robust optimization model for distribution network design under a mixed integer set of scenarios. *Computers & Operations Research* 136 (2021).

[7] Elizabeth D Dolan and Jorge J Moré. 2002. Benchmarking optimization software with performance profiles. *Mathematical programming* 91, 2 (2002).

[8] Genaro J. Gutiérrez, Panagiotis Kouvelis, and Abbas A. Kurawarwala. 1996. A robustness approach to uncapacitated network design problems. *European Journal of Operational Research* 94, 2 (1996).

[9] Johan Hellstrand, Torbjörn Larsson, and Athanasios Migdalas. 1992. A characterization of the uncapacitated network design polytope. *Operations Research Letters* 12, 3 (1992).

[10] Panos Kouvelis and Gang Yu. 2013. *Robust discrete optimization and its applications*. Vol. 14. Springer Science & Business Media.

[11] Thomas L Magnanti and Richard T Wong. 1984. Network design and transportation planning: Models and algorithms. *Transportation science* 18, 1 (1984).